Listing of the Claims:

This listing of claims will replace all prior versions, and listing of claims in the application:

1. (Currently Amended) A **computer readable medium including at least computer program code for a reduced** set of **load constant value** virtual machine instructions suitable for execution in a virtual machine to load constant values on an execution stack, the set of virtual machine instructions representing a ~~number of corresponding~~ **complete set of** Java **load constant value** Bytecode executable instructions that ~~are~~ ~~also suitable~~ **can be executed** ~~for loading~~ **to load** constant values on an execution stack execution in the virtual machine ~~to load constant values on an execution stack~~,

wherein the **reduced** set of the virtual machine instructions consists of a number of virtual machine instructions that is less than the number of ~~the corresponding~~ **instructions in the complete set of** Java **load constant value** Bytecode executable instructions, ~~and~~

wherein every one of the ~~corresponding~~ **instructions in the complete set of** Java **load constant value** Bytecode executable instructions can be represented by at least one of the virtual machine instructions in the **reduced set of load constant value** virtual machine instruction**s;** ~~set.~~

wherein the reduced set of load constant value virtual machine instructions consists of a first, a second, and a third instruction, the first instruction suitable for pushing one byte values on the execution stack, the second instruction suitable for pushing 4 byte values on the execution stack, and the third instruction suitable for pushing 8 byte values on the execution stack;

wherein the first instruction includes a code portion and a data portion which are both represented in a code stream in the virtual machine, and

wherein the second instruction includes a code portion and a data portion which are respectively represented in a code stream and in a data stream in the virtual machine.

2. (Cancelled)

3. (Cancelled)

4. (Currently Amended) A <u>computer readable medium as recited in claim 1,</u> ~~set of virtual machine instructions as recited in claim 3~~,

wherein the set is suitable to load N byte constant values on the execution stack, and

wherein N is a positive integer.

5. (Currently Amended) A <u>reduced</u> set of <u>store local variable</u> virtual machine instructions suitable for execution in a virtual machine to store local variables onto an execution stack, the <u>reduced</u> set of <u>store local variable</u> virtual machine instructions representing ~~a number of corresponding~~ <u>the complete set of store local variable</u> Java Bytecode executable instructions that are also suitable for execution in the virtual machine to store local variables onto an execution stack,

wherein the <u>reduced</u> set of <u>store local variable</u> virtual machine instructions consists of a number of virtual machine instructions that is less than the number of the <u>complete set of store local variable</u> ~~corresponding~~ Java Bytecode executable instructions, ~~and~~

wherein every one of the ~~corresponding~~ <u>instructions in the complete set of store local variable</u> Java Bytecode executable instructions can be represented by at least one of the virtual machine instructions in the <u>reduced set of store local variable</u> virtual machine instruction set,

<u>wherein the set of virtual machine instructions consists of a first instruction and a second instruction, the first instruction being suitable for storing 4 byte local variables onto the execution stack, and the second instruction being suitable for storing 8 byte local variables onto the execution stack, and</u>

<u>wherein the first or the second instruction includes a code portion and a data portion which are respectively represented in a code stream and in a data stream in the virtual machine</u>.

6. (Cancelled)

7. (Cancelled)

8. (Currently Amended) A ~~set of virtual machine instructions~~ **computer readable medium** as recited in claim 5,

wherein the set of set of virtual machine instructions is suitable to store N byte local variables on the execution stack, and

wherein N is a positive integer.

9. (Currently Amended) A **computer readable medium including at least computer readable code for a load array** virtual machine instruction suitable for execution in a virtual machine to load values from arrays on an execution stack, the **load array** virtual machine instruction representing ~~two or more~~ **the complete set of** Java **load array** Bytecode executable instructions that are **used** ~~also suitable~~ for loading values from arrays on the execution stack**; and**

wherein the load array virtual machine instruction is loaded on the execution stack with an array reference to the array that can be used to determine the array type, and an array index that indicates the size of the array;

wherein the load array virtual machine instruction is represented in a code portion of a set of code and data streams in the virtual machine and the array reference and index are represented in the data stream.

10. (Currently Amended) A **computer readable medium** ~~virtual machine instruction~~ as recited in claim 9, wherein the arrays can be an array of 1 byte values, or an array of 2 byte values, or an array of 4 byte values, or an array of 8 byte values.

11. (Currently Amended) A **computer readable medium,** ~~virtual machine instruction~~ as recited in claim 9, wherein a header of an array is read to determine the type of the array.

12. (Currently Amended) A **computer readable medium including at least computer readable code for a reduced set of store array** virtual machine instructions

suitable for execution in a virtual machine to store values located on an execution stack into arrays, the **reduced set of store array** virtual machine instruction representing ~~two or more~~ **the complete set of** Java **store array** Bytecode executable instructions that are ~~also suitable~~ **used** for storing values located on an execution stack into an array,

wherein each of the store array virtual machine instruction is loaded on the execution stack with an array reference to the array that can be used to determine the array type and an array index that indicates the size of the array, and

wherein each of the store array virtual machine instruction represented in a code portion of a set of code and data streams in the virtual machine.

13. (Currently Amended) A **computer readable medium** ~~virtual machine instruction~~ as recited in claim 12, wherein the arrays can be an array of 1 byte values, or an array of 2 byte values, or an array of 4 byte values, or an array of 8 byte values.

14. (Currently Amended) A **computer readable medium** ~~virtual machine load array instruction~~ as recited in claim 12, wherein a header of an array is read to determine the type of the array.

15. (Currently Amended) A **computer readable medium including at least computer readable code for a reduced set of duplicate value** virtual machine instruction suitable for execution in a virtual machine to duplicate values stored in an execution stack on top of the execution stack, the **reduced set of duplicate value** virtual machine instruction representing ~~two or more~~ **the complete set of** Java duplicate Bytecode executable instructions that are also suitable for duplicating values stored in the execution stack on top of the execution stack

wherein each one of the instructions in the reduced set of duplicate value virtual machine instruction has a parameter associated with it to indicate which value stored in the execution stack should be duplicated on the top of the stack; and

wherein each one of the instructions in the reduced set of duplicate value virtual machine instruction is represented in a code portion of a set of code and data streams in the virtual machine.

16. (Currently Amended) A <u>computer readable medium</u> ~~virtual machine instruction~~ as recited in claim 15, wherein values that can be duplicated on the execution stack are not limited to values that are within first, second, and third positions from the top of the stack.

17. (Currently Amended) A <u>computer readable medium</u> ~~virtual machine instruction~~ as recited in claim 15, wherein the values duplicated on top of the stack can be 4 byte values or 8 byte values.

18-20. (Canceled)

21. (Currently Amended) A <u>computer readable medium including at least</u> <u>computer readable code for a return value</u> virtual machine instruction suitable for execution in a virtual machine to return values by placing them on top of an execution stack, the virtual machine instruction representing ~~two or more~~ <u>the complete set of</u> Java <u>return value</u> Bytecode executable instructions that ~~are also suitable~~ <u>can be used</u> for returning values by placing them on top of the execution stack<u>, wherein the return value virtual machine instruction operates to read an index stored on the execution stack to determine the return value, and wherein the return value virtual machine instruction is represented in a code portion inside the virtual machine, and the index is stored in a data portion inside the virtual machine</u>.

22. (Currently Amended) A <u>computer readable medium</u> ~~virtual machine instruction~~ as recited in claim 21, wherein the values returned can be 4 byte values or 8 byte values.

23. (Currently Amended) A <u>computer readable medium including at least</u> <u>computer program code for an instantiate</u> virtual machine instruction suitable for execution in a virtual machine to instantiate Java objects and arrays, the <u>instantiate</u> virtual machine instruction representing ~~two or more~~ <u>the complete set of</u> Java <u>instantiate</u> Bytecode executable instructions that are also suitable for instantiation of Java objects or arrays<u>; and</u>

<u>wherein the instantiate virtual machine instruction operates to determine type based on a parameter value which is loaded on the execution stack; and</u>

wherein the instantiate virtual machine instruction is represented in a code portion of a set of code and data streams in the virtual machine.

24. (Cancelled)